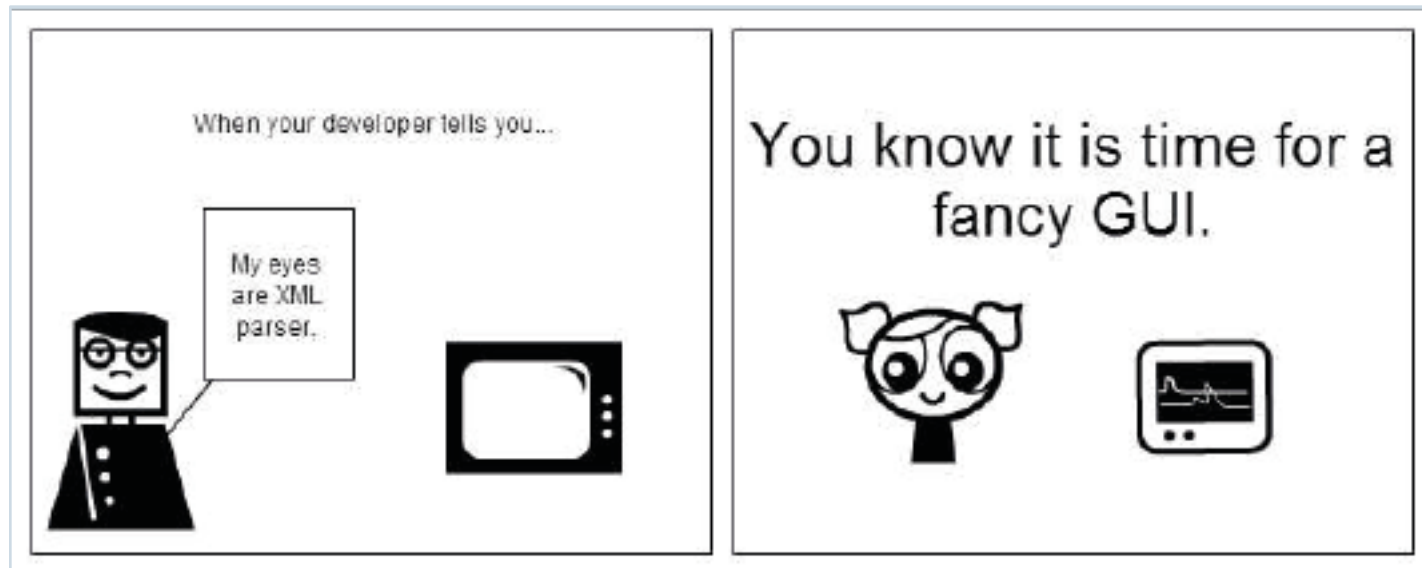


Visualizing DSI's Output

Force-Directed Layout with DSI-Guided Constraints for Visualizing Points-to Graphs

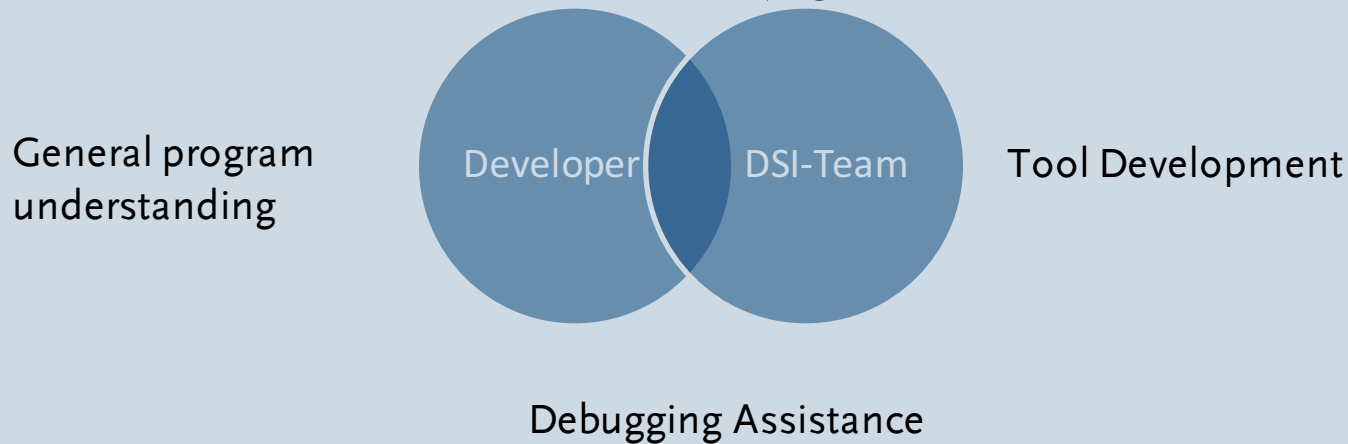


Why Visualization?



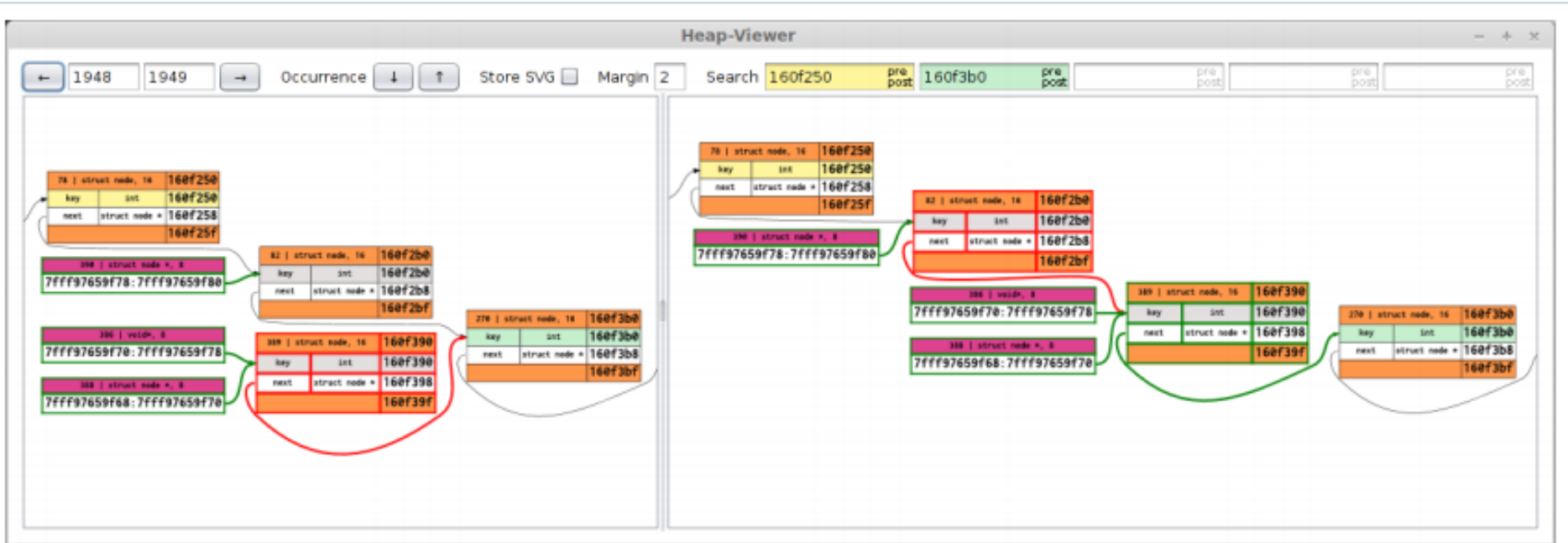
Why Memory Graph Visualisation ?

- What is a memory graph (points-to graph)?
 - “A memory graph gives a comprehensive view of all data structures of a program ...”²
- Who benefits from a memory graph?



- Why do we not have them in our IDE's?

Memory Graphs



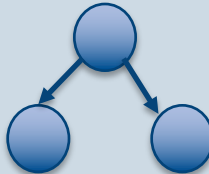
D.White, dsOli: Data Structure Operation Location and Identification

Desirable properties of Memory Graphs

...the layout should be:

- Tidy, no crossing edges, evenly distributed
- Emphasise data structures:

- A linked list: 

- A tree: 

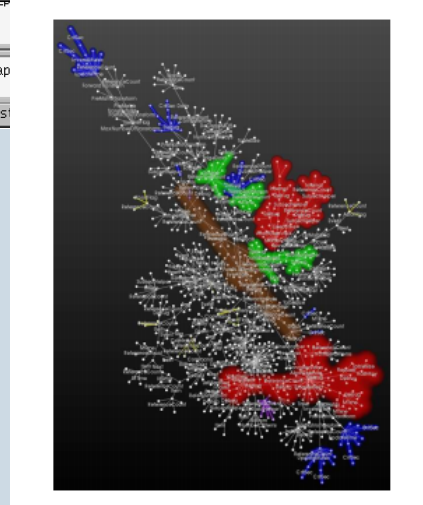
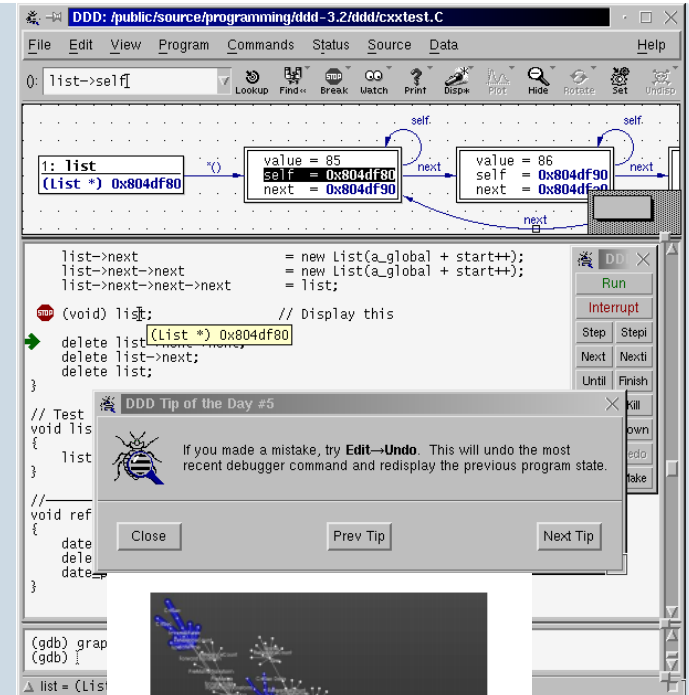
- Emphasis Levels of data structures and their relation

...the functionality should include:

- Show operations on the data structures
- Connect the objects of the representation with the source code
- Give the possibility of navigating through time steps

Is there a state of the art?

- Founders of GDB's offer DDD²
 - No special layout for data structures
- Experiments with landmarks³
- Experiments with collapsing data structures:
 - V.Singh, Raijv Gupta, Iulian Neamtiu: MG++: Memory Graphs for Analyzing Dynamic Data Structures
 - M.Marron, C. Sanchez, Z.Su, M. Friedrich: Abstracting Runtime Heaps for Program Understanding
 - E.Aftandilian et. al.: Heapviz: Interactive Heap Visualization for Program Understanding and Debugging



What is a force directed layout?

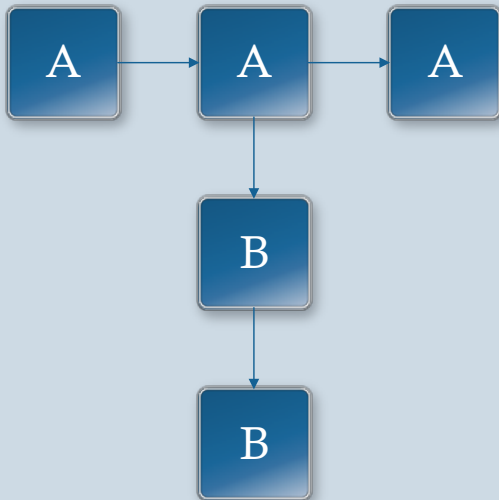
- A force directed layout...
 - Simulates a graph as a physical system
 - Assigns forces to nodes and edges
 - Run the simulation until an equilibrium is established
 - to position nodes, so that all edges are equal
- in d3 while each iteration of the simulation a function `tick()` is called
 - in this function we are able to influence the layout



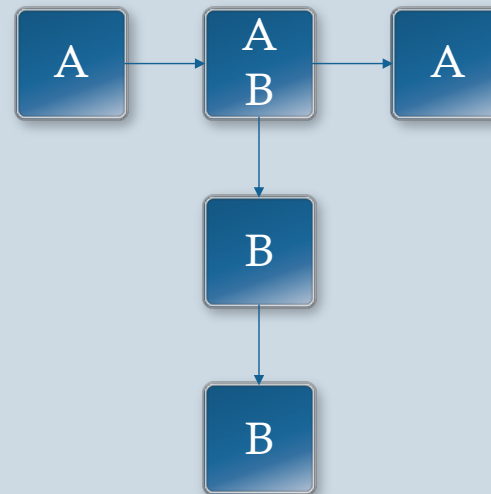
Constrained Based Layout: Challenge

- General Idea:
 - Pushing a force directed layout to draw groups of nodes in a predefined structure

Independent data structures

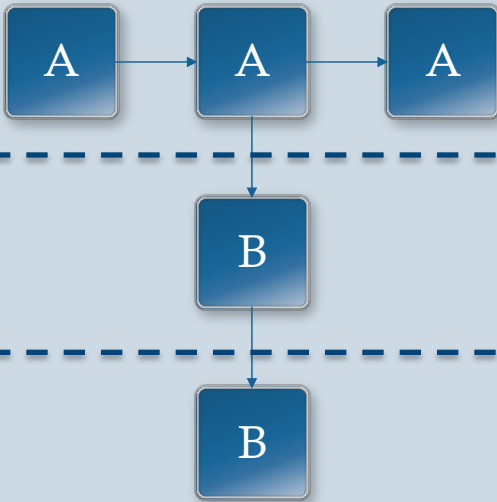


Overlay data structures

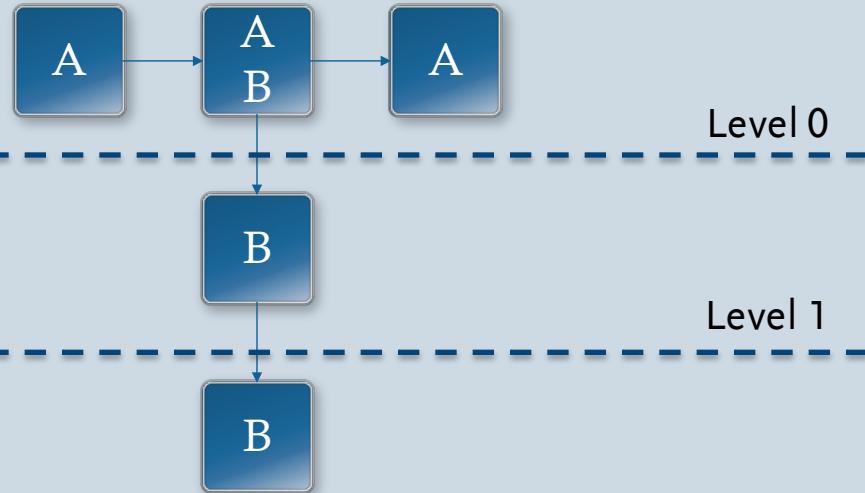


Constrained Based Layout:

Independent data structures



Overlay data structures



- Therefore we identified different steps:
 - Separate data structures which have no connection to each other
 - Separate overlay structures in component and anchor type
 - Annotate every data structure with level, structure type and anchor type information

Structure of dsOliViz



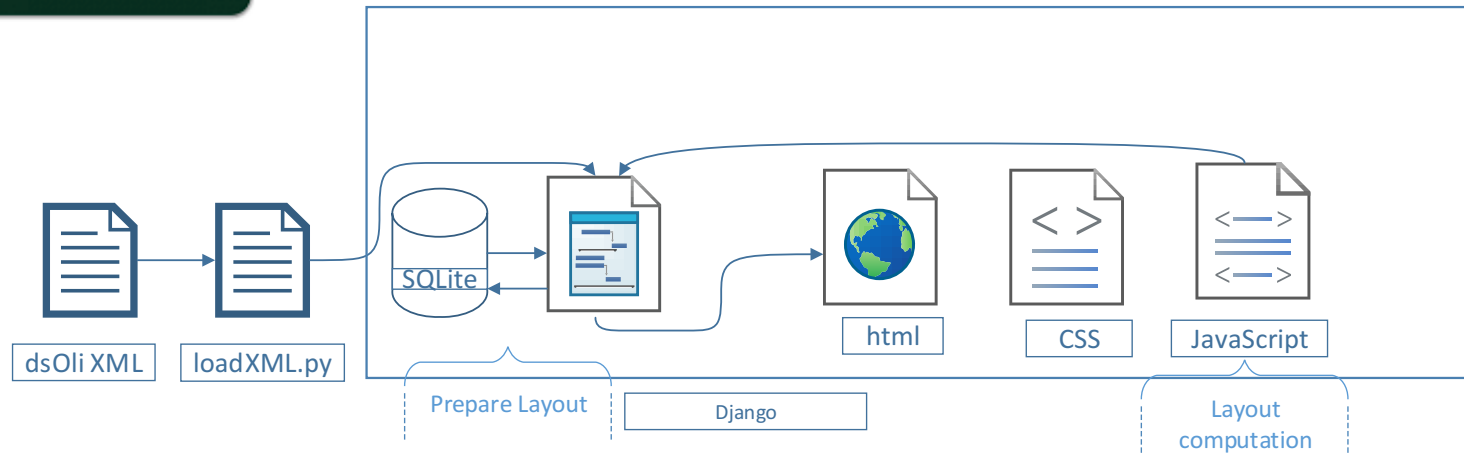
Data visualization in the web

Create graphs with SVG

Good object relation mapper

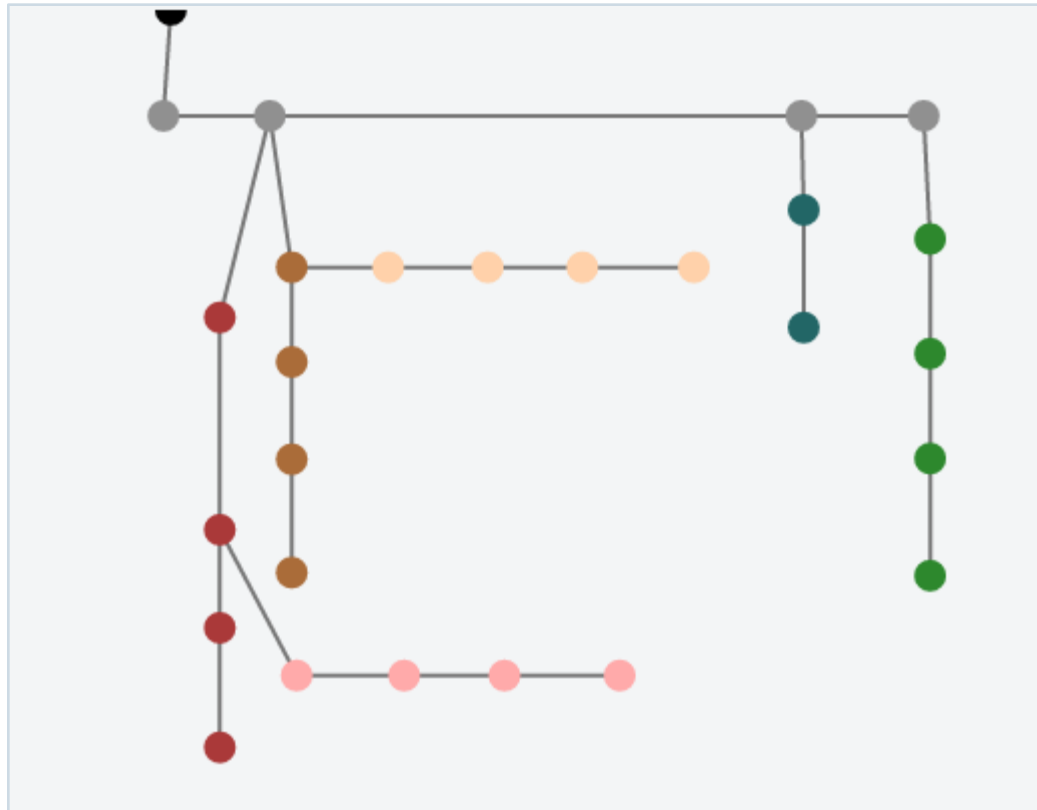


Automatic handling of Webserver



Responsive Design

Live Demo



Conclusion

- First tests have shown that constraints can push nodes in shape forms

Future Work:

- Connection to source code
- Collapsing of data structures
- Different views for developer, dsOli team member
- Search fields for strands/vertexes...
- Overview of relation between heap elements and time

Thank you for your for your attention.

Questions?

References

1. Thomas Zimmermann, Andreas Zeller: Visualizing Memory Graphs
2. Colin Myers, David Duke: A Map of the Heap: Revealing Design Abstractions in Runtime Structures
3. D.White, dsOli: Data Structure Operation Location and Identification
4. V.Singh, Raijv Gupta, Iulian Neamtiu: MG++: Memory Graphs for Analyzing Dynamic Data Structures
5. M.Marron, C. Sanchez, Z.Su, M. Friedrich: Abstracting Runtime Heaps for Program Understanding
6. E.Aftandilian et. al.: Heapviz: Interactive Heap Visualization for Program Understanding and Debugging